

文章编号:1003-7837(2005)01-0049-04

智能气门电镦机人机交互及串行通讯程序*

陈健,朱建伟,孙友松

(广东工业大学信息工程学院,广东广州 510643)

摘要:介绍了智能气门电镦机的数据显示和命令输入的人机界面(HMI)和实现电镦机实时控制功能的下位计算机之间利用RS232串行接口的通讯程序的编写,以及实现硬件定时的原理和方法.实践证明,该系统人机界面美观、实时性能可得到保证,符合气门电镦的工艺要求.

关键词:电镦机;RS232;编程;实时控制

中图分类号:TP31 **文献标识码:**A

在工业控制领域中,实时控制是很重要的.特别是在电镦机的工作过程中,镦粗速度、加热电流及镦粗压力等参数的实时控制尤为关键.作为一个单任务、单目标的操作系统,MS-DOS给工业系统实时控制软件的开发提供了一个非常好的平台.在DOS环境下,利用Turbo C或其他语言,可以很方便地开发出具有极高响应速度的实时控制软件.然而,要在DOS环境下设计出一个图文并茂、操作方便的用户界面,却是一件费时费力的工作.Windows操作系统的普及应用,尤其是可视化开发软件Visual C++的出现,为软件开发提供了强大的图形界面功能,使得人机交互、数据处理的图形化更为方便.

传统的电镦机的参数如镦粗速度、加热电流等都是根据经验人工调整,缺乏精确性和重复性,很难保证调整结果的一致性,导致生产的气门废品率很高.本文介绍的智能气门电镦机可以根据工艺自动调整控制参数,具有实时数据采集、实时控制和强大的人机交互能力、数据处理能力.智能电镦机的控制部分分为两级,下位机使用MS-DOS操作系统,主要负责实时数据采集及实时控制,上位机使用Windows 2000操作系统,主要负责人机交互和数据处理.

智能电镦机的自动化程度较高,在电镦过程中能够根据工艺自动调整控制参数,如镦粗缸速度、镦

粗压力及加热电流等,这些参数都必须实时地采集和控制.本文叙述了智能电镦机的上、下位机之间通过串口通讯来传送控制参数的初始值和电镦过程中的状态参数等.智能电镦机的上、下位机都使用工控机,其中下位机通过更改计数器/定时器卡上INT8的计数值进行定时^[1].

1 智能气门电镦机通讯、定时的实施^[2]

通讯及数据传递可以有多种方法实现.利用Visual C++(VC++)的标准通讯函数_inp和_outp可实现串口的读写.这两个函数的函数原型分别为:

int_inp(unsigned short port);

int_outp(unsigned short port,int databyte);

其中_outp()函数的第1个参数为端口地址,第2个参数是需输出的数据;_inp()函数的参数为地址变量,返回的是从该地址读取的数据.一般,COM1的默认端口地址为0x3F8,COM2的默认端口地址为0x2F8.也可以使用MS Comm串行通讯控件MS Comm32.OCX实现串行通讯,它提供了使用RS-232接口来进行数据通讯的所有协议.VC为该控件提供了标准的时间处理函数过程,并通过属性和方法提供了串行通讯的设置.它使用户能够

收稿日期:2004-12-08

*基金项目:广东省自然科学基金资助项目(990141)

作者简介:陈健(1964-),男,湖南汨罗人,副教授,博士.
万方数据

方便地访问 Windows 串行通讯驱动程序的大多数特性,包括输入、输出缓冲区的大小及决定何时使用流控制命令挂起数据传输等.此外还可以使用 API 函数.它是附带在 Windows 内部的一个极其重要的组成部分. Windows 的 32 位 API 主要是一系列很复杂的函数和消息集合,可以看作是 Windows 系统为在其下运行的各种开发系统提供的开放式通用增强功能接口.下位机则使用以下的函数对串口进行读写:

```
int inport( int portid );
unsigned char inportl( int portid );
void outport( int portid , int value );
void outportl( int portid , unsigned char value );
```

修改 INT8 进行定时,INT8 的时钟频率为 1.93180 MHz. 系统作为时钟使用时,每秒产生 18.2 次中断,可以设置这个时钟的计数值产生不同频率的中断.

首先设置计数器工作方式:

```
Outportl( 0x43 , 0x36 );
```

然后设置中断频率:

```
Outportl( 0x40 , l( counter ));
Outportl( 0x40 , h( counter ));
```

最后是保存和设置中断向量:

```
Intvector = getvec( 0x1c );
Setvec( 0x1c , realtimer_int_server );
```

其中 counter 为中断频率, realtimer_int_server 为中断函数的指针.注意:中断后要计算出此定时与实际时间的误差,在程序退出前要重新设置系统日历计数.

由于系统的工艺要求,此电测机实时控制系统规定的检测/控制周期为 20 ms. 串口通讯的速率设定为 19200 bps,即 2400 bytes/s = 48 bytes/20 ms. 然而这只是理论上的传输速率,实际应用中由于上位机使用多任务的 Windows 2000 操作系统,以及在 20 ms 的周期里还要运行检测、控制等其他的程序,所以实际允许的传输速率只有 30 bytes/20 ms 左右.当要求的传输数据量较允许的传输速率大时,就需要分时传送数据.

传送的数据可以是数字量也可以是字符.若使用数字量的格式进行传送,传送的数据量相对少得多,16 bit = 2 bytes,所以传送一个范围为 0~4095 的数字量只使用 2 bytes. 缺点是数据帧中较难使用命令字符,如使用字符 'a' 作为命令字符,而 'a' 的 ASCII 码为 97,与 0~4095 的范围重叠.可行的方

法是经过转换,把 'a' 字符对应 0~4095 范围外的数字,或者直接使用 0~4095 范围外的整数作为命令字符,这种做法的缺点是使程序的可读性变差.

若使用字符的格式进行传送,传送的数据量就会相应增大,传送一个范围为 0~4095 的数字量,例如 '4001',先把数字转换成 4 个字符 '4'; '0'; '0'; '1',传送的数据量为 4 bytes. 优点是可以直接使用字母作为命令字符,如 'a' 对应的 ASCII 码与数字 '0'~'9' 的 ASCII 码不会重叠.

2 编程实现

2.1 串口的读写

上位机使用 VC++ 6.0 的 MS Comm 串行通讯控件.每当计算机从串口接收到字符,会产生硬件中断,Windows 系统接收中断后会发出系统消息以调用以下函数:

```
void CHuajidiDlg : ChuanKou_OnMscmm1( )
//监听串口
```

在 ChuanKou_OnMscmm1()中读、写串口的操作:

```
VARIANT V1 ;
V1 = m_MsComm1.GetInput( );
//读取串口数据
CString str_temp ;
m_MsComm1.SetOutput( COleVariant
( str_temp )); //向串口写字符串
```

下位机使用 Turbo C++ 3.0 进行编程,使用 outportl(), inportl() 函数进行串口的读写,并把 outportb 函数包装成 send_char_com 函数.

```
void send_char_com( int portn , int cc )
{
outportl( portn , cc );
}
```

读取串口的操作利用了中断 11 和 12. 通过在中断函数中调用 inportl() 函数来读取串口接收的数据.

```
void interrupt com_in( ... )
{
inbu[ inhead++ ] = inportl( portbase +
RXR ); //read character into inbuffer
}
```

在设置中断向量之前要先保存原来的中断向量,程序结束时要先恢复原来的中断向量,否则影响系统的稳定:

```

void setvects( void )
{
    oldvects[ 0 ]= getvec( 0x0B );
    oldvects[ 1 ]= getvec( 0x0C );
    setvec( 0x0B , com_int );
    setvec( 0x0C , com_int );
}

void resvects( void )      //恢复中断向量
{
    setvec( 0x0B , oldvects[ 0 ] );
    setvec( 0x0C , oldvects[ 1 ] );
}

```

2.2 分时传送数据的实现

本控制系统使用字符的格式进行数据传送,系统运行中需要传送的任务为:从下位机传送 16 组数据到上位机,每组数据是一个 0~4095 的数字量,转换成长度为 4 bytes 的字符串,这样共有 64 个字符。由于在一个 20 ms 的周期内最多只能传送 48 个字符,所以数据的传送要分组传送。

上述 16 组数据中的镦粗缸和砧子缸的位移参数是用来计算工件长度的,要求的精度高,所以要保证每个周期都传送这两个参数。因此分成五个数据帧,每个帧都包含镦粗缸和砧子缸的位移参数,而且都有一个 1 至 5 的序号,作为接收端分辨帧的依据。

数据帧的格式为:“h”+“g”+“数据”+“校验码”+“e”

“h”帧头,表示一个数据帧的开始;“g”:帧类型,根据帧的序号依次为 g1、g2、g3、g4、g5;“数据”:数据内容,字符串,是由 0~4095 的数字量转换成的字符组成;“e”:帧尾,表示数据帧的结束。

校验方法为异或校验,校验码长度为 1 byte,用 00000000B 与“头字节”、“帧类型”和“数据”进行异或得出校验码。接收方使用校验码对“头字节”、“帧类型”和“数据”进行异或。若结果为 0 表示接收正确,否则为接收错误,丢弃该帧。

2.2.1 下位机发送数据的程序

数字量转换为字符串:

```

for( i=0 ; i<4 ; i++ )
{
    int_temp[ i ]= int_input % 10 ;
    int_input = int_input / 10 ;
    char_temp[ i ]= int_temp[ i ]+ ' 0 ' ;
} // 万方数据

```

发送字符串:

```

for( i=0 ; i<4 ; i++ )
{
    send_char_com( char_temp[ i ] );
}

```

2.2.2 上位机接收数据的程序

在 CHuaijiDlg::ChuanKou_OnMscomm1() 函数中:

变量的定义:

```

VARIANT V1 ;
CString str_V1 , str_V2 ;

```

接收帧:

```

m_MsComm1.SetInputLen( 1 );
V1 = m_MsComm1.GetInput( );
str_V1 = V1.bstrVal ;
if( * str_V1 == ' g ' )
{
    m_MsComm1.SetInputLen( 2 );
    V1 = m_MsComm1.GetInput( );
    str_V1 = V1.bstrVal ;
    Switch( str_V1 )
    {
        case " g1 " : ..... ;
                        break ;
        case " g2 " : ..... ;
                        break ;
        case " g3 " : ..... ;
                        break ;
        case " g4 " : ..... ;
                        break ;
        case " g5 " : ..... ;
                        break ;
    }
}

```

V1 是 VARIANT 型数据,是 MsComm::GetInput() 函数返回参数的指定类型。可以使用 V1.bstrVal 这种方式取得 V1 包含的字符串。

MsComm::SetInputLen(int) 是设定从串口缓冲区中读取的字符数。程序中的

```

m_MsComm1.SetInputLen( 2 );
V1 = m_MsComm1.GetInput( );

```

表示读取两个字节,并把返回的字符串赋给 V1。

使用 Switch(str_V1) 语句判断数据帧类型,在 case 语句里调用相应的程序接收相应的帧数据。

3 结 论

使用一种外部定时/计数器板卡来进行定时. 串口通讯是传统的通讯方式,特点是简单易用.虽然在实时控制中应用串口通讯具有一定的困难,但在传输数据量不是很大的情况下,实时控制下的串行通

讯还是可靠的.实践证明该系统能够满足气门电镦的工艺要求,生产出合格的产品.

参考文献:

- [1] 陈健,张忠夫,孙友松,等.气门电镦机计算机控制系统研制[J].机床与液压,2004,(6)34-36.
[2] 陈健,朱建伟,孙友松.实时控制下的电镦机镦粗动作模拟[J].锻压装备与制造技术,2004(5)33-35.

Human machine interaction and serial communication program between the host computer and the real time controller for an intelligent electric valve upsetting machine

CHEN Jian, ZHU Jian-wei, SUN You-song

(Faculty of Information Engineering, Guangdong university of technology, Guangzhou 510651, China)

Abstract: Programming of the serial communication between the host computer and the real time controller, which is used to transfer data and system parameters is described in detail in this paper. And an timer/counter card is introduced to give an exact timing of the real time controller. Practice proves that this system is of beautiful human computer interface and assured real-time property, meeting the technology demand for valve upsetting.

Key words: electric upsetting machine; RS232; programming; real time control



GYX 系列高频振动细筛

GYX 系列高频振动细筛是细粒物料的筛分设备,广泛应用于矿山、冶金、煤炭、化工、建材、食品及轻工等行业.我们生产 GYX 系列产品规格齐全,亦可根据用户需求生产各种规格设备.

* 设备特点:处理量大,筛分效率高,分离粒度细,湿、干式筛分粒度均可达 40 μm .振动频率高,振幅小.振动筛电机性能优良、寿命长,管理、维护方便.

筛体采用橡胶弹簧悬挂式进行减销振,机架振动小,无须固定,噪音低,橡胶弹簧寿命长.采用多层筛网重叠技术,有效地防止筛网堵塞.使用更换筛网的专用工具,筛网更换方便、快捷.

* 规格:GYX11-1010、GYX31-1007、GYX31-1207、GYX21-1210.

* 地址:广州市天河区长兴路广州有色金属研究院 邮政编码:510651

网址:<http://www.gzxks.com>

电话:020-37239220 37239221 37239030

传真:020-37238535 E-mail: xks@mail.gzrnm.com

